Remarks

Claims 1-10, 13-30, 33-40, and 59-72 are pending in the application. All claims stand rejected. By this paper, claims 1, 2, 9, 20, 21, 29, 30, and 40 have been amended. Claim 59 has been cancelled.

Claims 1-6, 10, 13-26, 30, and 33-40 were rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,490,547 to Atkin et al. ("Atkin") in view of U.S. Patent No. 6,275,978 to Bell ("Bell"). This rejection is respectfully traversed.

Claim 1 has been amended to clarify the issues raised by the applicants in paper #16. As amended, claim 1 recites a system for providing multiple language support for at least one application program, comprising:

> a plurality of language resource bundles comprising associations between language keys and displayable language-sensitive elements, each resource bundle corresponding to a different language, wherein at least one association is specific to a particular application program and at least one association is applicable to a plurality of different application programs; and
>
> a language resource manager configured to receive a first language key from a first application program, locate a language resource bundle corresponding to a currently-selected language, identify a language-sensitive element associated with the first language key and the first application program, and provide the identified language-sensitive element to the first application program for display in a graphical user interface.

These claimed features allow associations between language keys and language-sensitive elements (*e.g.,* translations) to be application program-specific, *i.e.*, different language keys may have different meanings depending on which particular application program is accessing them. For instance, the term "page" may have one meaning for a Web browser, *e.g.,* a "Web" page, while meaning an entirely different thing to a word processor, *e.g.,* a page of a word processing document. Accordingly, the term "page" may need to be translated differently for each application program, requiring the creation of an application program-specific association between a language key and language-sensitive element. On the other hand, certain terms may have uniform meanings across applications, such as "start," "exit," "file," etc. In such a case, an association that is not application program-specific may be created.

Atkins does not disclose application program-specific associations. Indeed, once Atkins' text string translation engine 212 has provided a translation for a particular word, that translation is used consistently for all future occurrences of the word. Thus, Atkins actually teaches away from the concept of application program-specific associations.

The addition of Bell does not cure the deficiencies of Atkins. Bell discloses a system for differentiation of localized terms. As explained by Bell, an example of differentiation relates to "the string 'hot' which in English refers to situations for both temperature and taste. A localization for these two different cases in Spanish have two different meanings[:] The 'picante' (taste) and 'caliente' (temperature)." Col. 1, lines 58-62 (emphasis added). To differentiate between these cases, Bell inserts a second key to indicate which translation should be used so that the proper meaning of "hot" is conveyed.

Thus, Bell's associations are, at best, case or context-specific, not application program-specific, as required by amended claim 1. There is no teaching or suggestion in Bell, without hindsight reliance upon the applicants' own teachings, for making Bell's keys specific to a particular application program.

Furthermore, there is no teaching or suggestion in Bell for identifying a "language-sensitive element associated with the first language key and the first application program," as required by claim 1. At best, Bell looks for a second key to determine the proper context of a translation. Bell does not select a particular language-sensitive element based on (1) a first language key and (2) the first application program, which does not require a second key as in Bell. A system in accordance with the claimed invention may simply look at which application program is making the request.

Even if the references are combined, the combination does not result in the claimed invention. Neither reference discloses application program-specific associations between language-keys and language-sensitive elements. At best, such a combination would produce the just-in-time localization system of Atkins with Bell's provision for a second key to indicate context, *i.e.*, "picante" (taste) vs. "caliente" (temperature), within a single application program.

In view of the foregoing, the applicants respectfully submit that claim 1, as amended, is patentably distinct over the cited references. Claims 21 and 40 have been amended to include similar limitations. Claims 2-10, 13-20, 22-30, and 33-39 depend directly or indirectly on claims 1, 21, and 40. Accordingly, the applicants respectfully submit that claims 1-10, 13-30, 33-40 are patentably distinct over the cited references, alone or in combination.

Claims 7-9 and 27-29 were rejected under 35 U.S.C. 103(a) as being unpatentable over Atkin in view of Bell and further in view of U.S. Patent No.

6,469,713 to Hetherington et al. ("Hetherington"). This rejection is respectfully traversed.

Claim 29 recites the steps of:

> preempting the first application program;
>
> saving a state of the first application program;
>
> discarding the graphical user interface being currently displayed;
>
> generating a new graphical user interface comprising at least one new language-sensitive element received in response to the language key;
>
> restoring the state of the first application program; and
>
> resuming execution of the first application program.

According to the Examiner, Hetherington teaches "a method for dynamic language switching wherein user interface dialogs may reload the contents of displays, updating the user interface display to contain the contents of menu labels, help text in the new human language or display text formatted in accordance with the new cultural convention."

However, Hetherington updates his user interface in a very different manner from that recited in claim 29. According to Hetherington, subscribing applications are registered as "listeners for such language, locale, and display change system messages. ... The system message may originate from a control dialog or from another application, and may be passed from one application to another or to dialogs associated with the receiving application. User interface dialogs or applications notified of the system message may reload the contents of displays, updating the user interface display to contain the contents of menu labels, help text, or dialog messages in the new human language or display text formatted accordance with the new cultural convention." Col. 2, lines 26-37.

Thus, Hetherington's applications are listeners, waiting for a system message to indicate a change in language. Once the message is received, it is up to the applications, themselves, to reload the contents of displays, etc.

The process recited in claim 29 is very different. The application programs do not listen for system messages and then reload content displays as appropriate. Rather, an application program is actively preempted, at the operating system level. The state of the application program is then saved by the system, after which the graphical user interface of the program is forcibly discarded and a new a graphical user interface is created comprising at least one new language-sensitive element received in response to the language key. Finally, the state of the application program is restored, after which normal execution is resumed.

Thus, Hetherington's updating is performed by the applications, while the claimed updating is imposed upon the applications. Moreover, Heatherington's applications must be specially configured to listen for system messages regarding language changes. An application program in accordance with claim 29 need not be specially configured, but may be any standard program including language-sensitive elements.

There is no teaching or suggestion in the references for preempting a first application program or saving the state of the first application program, as required by amended claim 29. Indeed, there is no teaching or suggestion of imposing a change of language upon an application that is already currently executing.

Accordingly, the applicants respectfully submit that claim 29, as amended, is patentably distinct over the cited references. Claims 9, 60, 63, and 66 include similar limitations and are thus believed to be patentably distinct for at least the same reasons.

In view of the foregoing, the applicant respectfully submits that claims 1-10, 13-30, 33-40, and 60-72, as amended, are patentably distinct over the cited references, alone or in combination. Early allowance of all pending claims herein is respectfully requested.

Respectfully submitted,

Park City Group, Inc.

By ________________________
Kory D. Christensen
Registration No. 43,548

STOEL RIVES LLP
One Utah Center Suite 1100
201 S Main Street
Salt Lake City, UT 84111-4904
Telephone: (801) 328-3131
Facsimile: (801) 578-6999